Remote Operators Prefer Goal-based Semi-Autonomous Algorithms

Shawn Hunt, Ph.D., Gary Witus, Ph.D., R. Darin Ellis, Ph.D., Abhilash Pandya, Ph.D.

UNCLASSIFIED: Dist. A. Approved for public release

Abstract: This focus of this research was to determine if reliable goal-based semi-autonomous algorithms are able to improve remote operator performance or not. Two semi-autonomous algorithms were looked at: visual servoing and visual dead reckoning. Visual servoing uses computer vision techniques to generate movement commands while uses internal properties of the camera combined with sensor data that tell the robot its current position based on its previous position. This research proved that the semi-autonomous algorithms developed increased performance in a measurable way. An analysis of tracking algorithms for visual servoing was conducted and tracking algorithms were enhanced to make them as robust as possible. The developed algorithms were implemented on a currently fielded military robot and a human-in-the-loop experiment was conducted to measure performance.

Keywords: Visual tracking, teleoperation, visual servoing, UGV

1. Introduction

Mobile robots, or Unmanned Ground Vehicles (UGVs), play an increasing role in both the defense and security of our nation and in the ability to respond to emergency situations. Robots have been used in Iraq and Afghanistan for bomb disposal. They also played a key role in searching for victims of the World Trade Center attack. They were created to keep our soldiers, or warfighters, out of harm's way.

The current method of UGV control is rate control teleoperation – is burdensome [1]. Figure 1 depicts the current way the robots are controlled. There is a high workload that requires constant attention and limits situational awareness. A

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **18 APR 2011**	2. REPORT TYPE **N/A**	3. DATES COVERED **-**
4. TITLE AND SUBTITLE **Remote Operators Prefer Goal-based Semi- Autonomous Algorithms (PREPRINT)**		5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) **Shawn Hunt Ph.D; Gary Witus Ph.D; Darin Ellis Ph.D; Abhilash Pandya Ph.D**		5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **US Army RDECOM-TARDEC 6501 E 11 Mile Rd Warren, MI 48397-5000, USA**		8. PERFORMING ORGANIZATION REPORT NUMBER **21756**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) **US Army RDECOM-TARDEC 6501 E 11 Mile Rd Warren, MI 48397-5000, USA**		10. SPONSOR/MONITOR'S ACRONYM(S) **TACOM/TARDEC/RDECOM** 11. SPONSOR/MONITOR'S REPORT NUMBER(S) **21756**
12. DISTRIBUTION/AVAILABILITY STATEMENT **Approved for public release, distribution unlimited**		
13. SUPPLEMENTARY NOTES **Submitted for publication in the Journal of Intelligent and Robotic Systems, The original document contains color images.**		
14. ABSTRACT		
15. SUBJECT TERMS		

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**SAR**	**18**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

dedicated operator is not able to perform multiple tasks and control of the UGV can be difficult when the terrain is rough or communications are degraded.

Figure 1: The current method to control a UGV is with a laptop using rate-controlled teleoperation.

These robots are currently being used in countries where there is an ongoing war. Those who have attempted to view a laptop's display while out on a sunny day can attest to how difficult it can be to view the contents of the screen. Add to that scenario a stressful situation of using the laptop trying to find a bomb buried in the soil and that paints a vivid portrait of why this research is needed and important to the Army. If all the warfighter has to do is designate a point for the robot to go to and they know that it will go to that point reliably, then their job becomes easier.

Robots have been in the news in recent months due to the BP oil spill in the Gulf of Mexico [2]. The robots were remotely controlled by BP personnel to try to cap the damaged oil well. BP ran into a setback to their containment efforts when a saw blade the robot was using became stuck [3].

The motivation behind this work was to provide a level of autonomy to existing robots used in the field so that operating a robot does not require constant supervision. The costs associated with developing fully autonomous system may potentially outweigh the benefits [4]. The recent series of DARPA Grand Challenges prove that fully autonomous robotic systems is indeed possible but technology that creates autonomous systems has at the same time also created unwelcome "automation surprises" [5]. DARPA is the military's research organization. It stands for Defense Advanced Research Projects Agency.

The Three Mile Island disaster in 1979 was caused by a system functioning on its own, attempting to compensate for a stuck valve. The operators of the nuclear power plant did not have sufficient time to act before the automated system transferred control to them in order to avert the disaster. The same problem occurs in the auto-pilot control in airplanes. If there are any problems with the system, they are often not communicated to their human operators in sufficient time to take proper action prior to system failure.

Situational awareness (SA) is also an important area of study, and although it isn't studied in-depth in this research, the work developed here provides a framework to study the effects that the semi-autonomous algorithms described in this thesis have on situational awareness. In [6, 7], Endsley broke SA into three levels, stating that it is being able to perceive elements in the environment, understand what all of the elements mean, and be able to project their status in the future.
The two semi-autonomous algorithms that are the focus of this paper are visual servoing and visual dead reckoning.

The military has been embracing the use of robotics in recent years to keep warfighters out of harm's way. In 2007, the Department of Defense released a roadmap [8] for the next 25 years, detailing its paradigm shift in fighting wars with robots. The roadmap also elaborated on a series of goals that the Department of Defense wants to achieve for its unmanned systems. These goals include:

- Improving the overall effectiveness of the unmanned systems through collaboration

- Achieving greater commonality and interoperability of unmanned systems
- Developing standards that support the safe operation and integration with the manned systems
- Using rapid prototyping and deployment to get the technology out to the warfighters as quickly as possible.

Although the roadmap was published several years ago, teleoperation remains the main method of control for fielded robots. This research added supervised autonomy to a military robot that is current used in missions. It is also important to note that the focus of this research is solely on unmanned ground vehicles (UGVs). Although there are natural extensions to underwater robots, that is not the focus of this paper.

Beyond the research questions of this work, in order for this to have wide acceptance by warfighters in the field, the system has to be easy to use and easy to learn. It should require less mental workload with it than without it. It should not require constant attention. It should be able to be given a destination and the operator knows that it will go there without fail. If it is burdensome to use, it will not be an acceptable form of control.

2. Implemented Algorithms

There were two algorithms that were implemented for this work: visual servoing and visual dead reckoning. Visual servoing [9] is simply the name given for using data captured from a camera to control the motion of a robot using computer vision techniques. The first papers published on visual servoing date back to the 1970s [10]. This has grown into a very large field of study [11] with many papers published. The papers have traditionally fallen into two broad categories: 2D, or image based (IB) [12], and 3D, or position based (PB) [13].

In position based control, image features are extracted and a model of the scene image features is used to estimate the pose of the target with respect to the camera using a geometric model of the target [14]. This approach is typically referred to

as 3D visual servoing in literature. This method requires precise calibration of the camera for it to be accurate.

Camera calibration [13] is the process of finding the camera parameters that affect the imaging process. Intrinsic camera parameters do not change for a particular camera-lens combination. Intrinsic camera parameters include the exact center of the image, the focal length, the lens distortion, and the scaling factors that are used for row and column pixels. The extrinsic camera parameters describe the camera's pose, or its position and orientation, in the world coordinate system. In [15], a methodology was published for autonomously calibrating a camera. Once the intrinsic and extrinsic camera parameters have been found, the pose of the camera in the workspace is able to be computed.

The second class of visual servoing algorithms is image-based [16]. In image-based visual servoing, the pose estimate is omitted [17] and the motion control is done solely in image space. There has also been work published on "2-1/2D" visual servoing [18] that bridges the two groups by trying to minimize the errors in the image and pose space.

The two visual servoing algorithms that were implemented for this system were a correlation-based tracker and an affine-based KLT tracker. Both algorithms belong to the image-based class of visual servoing algorithms.

Dead reckoning [19] has its roots outside the realm of robotics but it is basically estimating one's current position based on a previously determined position and advancing that position based on known speeds over time. Dead reckoning has been shown to be used in nature [20]. Dead reckoning has also been shown to be used in marine, air, and automotive navigation and it has even been proven to be successful in predicting latency and reducing its impact on networked games [21].

Visual servoing to control a robot has been an active area of research for many years. Purely relying on visual features can fail when the robot is operating in an environment with no features (e.g. a concrete floor with white walls). Visual dead

reckoning is a novel approach that was developed in this research. It uses odometry (as calculated by the Inertial Measurement Unit) along with kinematics of the arm. Visual dead reckoning first rotates then translates to the goal point.

First, for rotation, if the initial goal point is defined as C_0, and C_m denotes the middle of the image, then the pixel distance to rotate is given in (1). Next, if s is the horizontal IFOV, then how far the robot has to translate is given by (2).

$$C0 - Cm \tag{1}$$

$$s * (C0 - Cm) \tag{2}$$

Once the robot has finished rotating, it begins translating to the goal. The angle between vertical and the bottom of the image is given as A_b and may be calculated by (3), where s is the vertical IFOV (Instantaneous Field of View). Next, if the row position of the initial goal point is given as R0, then the stopping row is given as Rm, and H equals the height of the camera as determined by the forward kinemtatics, then the initial distance is given by (4). Once this value has been found, the odometry is used to determine when the goal point has been reached.

$$s * \frac{rows}{2} \tag{3}$$

$$H * \tan((Ab + s * R0) - \tan(Ab + s * Rm)) \tag{4}$$

3. Human-in-the-Loop Test

The goal of the human-in-the-loop test was to see how the supervisory control algorithms performed relative to teleoperation at different levels of dropout. The test used the robot shown in Figure 2.

Figure 2: The robot that was the focus of this research.

Participants

There were six participants that were all university students. All of the participants had normal/corrected vision. No subject had any cognitive impairment. All subjects had prior experience using a computer and playing video games.

Course Design

Three courses were constructed that looked similar to what is shown in Figure 3. The first course was made out of masking tape applied to the floor. The second course was designed to simulate small bumps and was made out of 1x2s as the bumps, with 2x4 as the rails. The third course simulated large bumps and was made out of 2x4s as the bumps

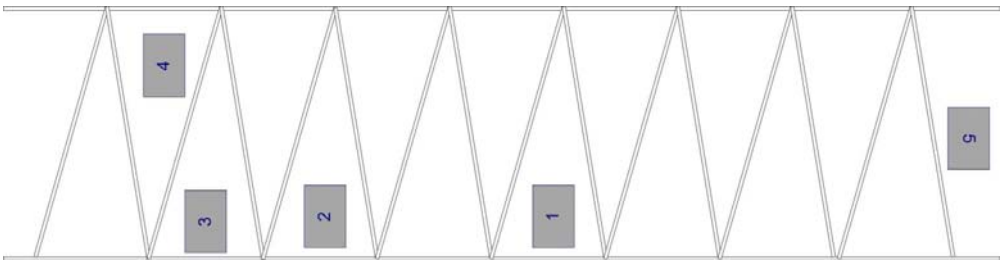

Figure 3: The layout of the courses with the five inspection targets.

The course was designed to have the robot traverse and come back. There were four stops on the down portion of the track. The four stops on the down portion of the track were approximately 22, 11, 5, and 2 feet apart. The fifth stop, going back to the starting position, had a distance of 45 feet. Figure 4 shows an overhead view of the three courses that were created in the laboratory. Figure 5Figure 5 is another view of the robot going over the 2x4 course.

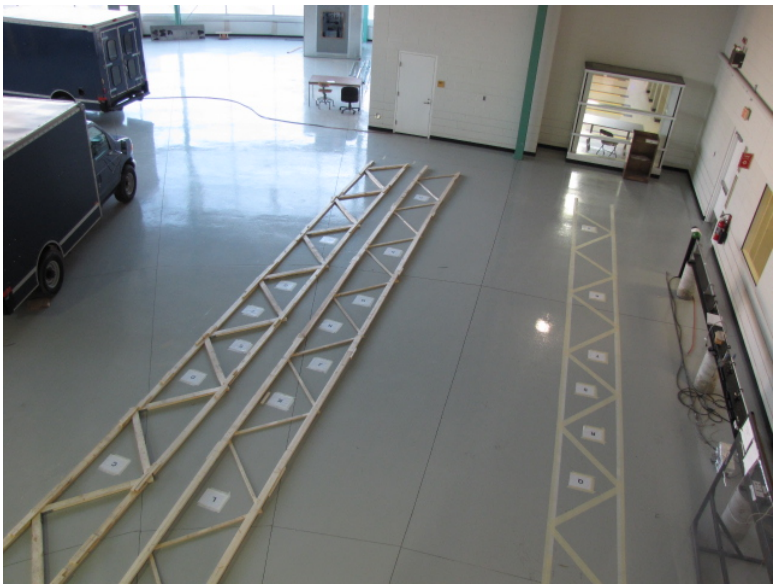

Figure 4: The overhead view of the three courses: flat (with tape), 1x2s, and 2x4s.

Figure 5: The robot going over the 2x4 course during subject testing.

Inspection Tasks

Each participant was asked to move the robot to each of the five targets using each of the different control methods while keeping the robot inside the rails. Once the participant felt that the robot was positioned correctly, they were instructed to take a snapshot. A running count of how many snapshots had been taken was shown in the upper-left corner of the screen. This was done in order to provide an easy method of analyzing the data after the experiments were done. The post-processing process looked at the timestamp of when the subject took the picture to break each run up into the different segments. There were several occasions in the data files where this did not happen and the subject took too many snapshots by accident. In these cases, the odometry data were used to determine when the robot started to move, indicating a new goal point was issued. For the supervisory control algorithms, the participant was told that they could issue an emergency stop to the robot at any time by pressing the space bar on the keyboard but that the goal was not to rely on it because ideally, both visual servoing and visual dead reckoning should go exactly where the subject clicked. If the subject had to press the emergency stop many times, that indicated that the semi-autonomous algorithms were not performing well.

Dropout Rates

Simulated degraded communications were introduced by corrupting data packets. A corrupted data packet is one that cannot be decoded. The data containing control messages from the OCU to the robot and data packets containing the video feed from the robot to the OCU were both artificially corrupted. In the current fielded system, when a corrupted video packet is received, a black frame is shown. In this implementation, the last good frame was displayed. The data corruption was modeled as a Bernoulli process, i.e. all of the packets had an equal probability of being corrupted. There were four levels of communication degradation that were implemented: 0, 3/8, 9/16, and 3/4 seconds.

Trial Procedure

Prior to beginning, each subject was given the same presentation that detailed the objectives of the study. A graphic of the course was first used to explain where

the target locations were and the course was also walked with the subject to show where each target was located. The subject was given ample time to use each control method before the actual test and indicated to the test proctor when they were comfortable enough to proceed.

The subject was positioned in an area that had no direct line of sight to the robot, as shown in Figure 6. Each subject was instructed not to turn around and look at the robot while they were controlling it. At the end of a course run, each subject was asked to enter a difficulty rating on a one to ten scale, when one meant easy and ten meant difficult. This provided the examiner with a difficulty rating for each control method, dropout rate, and course roughness.

Figure 6: Another view of the experiment in the laboratory. Each participant was positioned in such a way that the robot could not be seen.

There were times when the robot would become unresponsive due to communication interference or it would stop because of discharged batteries. If this occurred during a trial run, the trial was repeated.

The subjects were asked to complete a task, namely to drive to a target on the ground and stop the robot when the target is still visible in the display and is within reach of the robot arm. This position was chosen because the target was in reach of the grippers on the robot arm. In a realistic setting, this would be similar to driving up to something buried in the ground that a warfighter wants to examine with the robot.

Experimental Design

The experiment was run with six subjects. For each subject, the test was blocked by control method: teleoperation, visual dead reckoning, visual servoing using the correlation tracker, and visual servoing using the KLT tracker. Each of these blocks was then subdivided into four blocks by the dropout rate. Each of these blocks consisted of runs on each of the three sources. Each subject ran a total of 48 courses for a total of 288 course runs over each of the six subjects. Each subject took between 4 and 6 hours to complete all runs and each subject completed the test in a single block of time, i.e. no one came back at a later date to complete the test.

Data Validation

The first pass of the data occurred before the subject left. This test made sure that the data files had all been properly recorded. The data parsing program used the timestamps of when the operator took the picture when the operator felt they were positioned correctly. The operator would sometimes accidentally press the button too many times. If the data reduction program ran into this scenario, it would automatically try to combine the timestamps based on movement of the robot. There was a field in the reduced file that indicated when this happened so that the result could be manually verified to make sure nothing was lost.

The difficulty scores from of each of the runs were stored separately from data collected from the robot. These had to be combined at data reduction time. An inspection was made of each record to make sure that the difficulty ratings from the database were brought over correctly in the final file. The reduced file was also visually inspected to make sure all of the fields were within the normal range (i.e. the angles from the IMU readings were all between 0 and 360 degrees).

Testbed Limitations

The testbed described does have limitations. The environment tested was relatively benign because there were no obstacles that the operator had to navigate around other than the simulated rough terrain. The next step would to add

obstacles in the robot's path, followed by creating a test plan for a relevant outdoor environment.

4. Results

This was a within-subject design with subjects used as replicates. The values for the course roughness in the supplemental figures and tables are: 0 = flat, 1 = 1x2 course, and 2 = 2x4 course. The values for control method are: 0 = teleoperation, 4 = visual dead reckoning, 5 = correlation and 6 = KLT. The values for the dropout rate are: 0 = no delay, 1 = 3/8, 2 = 9/16, and 3 = 3/4.

After each run, the participant was asked to rank the difficulty on a scale from 1 to 10, where 1 meant easy and 10 meant difficult. Table 1 shows the results of a two-way ANOVA of difficulty rating as a function of dropout and course roughness ($F_{2,1716} = 24.73$, p=0.0000). The difficulty rating increased as the roughness of the terrain increased. The difficulty rating increased as the dropout rate increased as well. There was no significance in the interaction between dropout rate and course roughness. Figure 7 shows the box plot of the two-way ANOVA results. Figure 8 shows the mean values of the difficulty ratings by dropout rate and course roughness. The 3/4 dropout rate had the highest average difficulty rating at 5.229 and the 2x4 course had the highest average difficulty rating at 5.339.

Two-way ANOVA: difficulty_rating_1easy_ versus course_number, dropout_rate

Source	DF	SS	MS	F	P
course_number	2	295.1	147.536	24.73	0.000
dropout_rate	3	173.6	57.875	9.70	0.000
Interaction	6	15.0	2.502	0.42	0.867
Error	1716	10238.2	5.966		
Total	1727	10721.9			

S = 2.443 R-Sq = 4.51% R-Sq(adj) = 3.90%

Table 1: Two-way ANOVA results of difficulty rating as a function of course roughness and dropout rate.

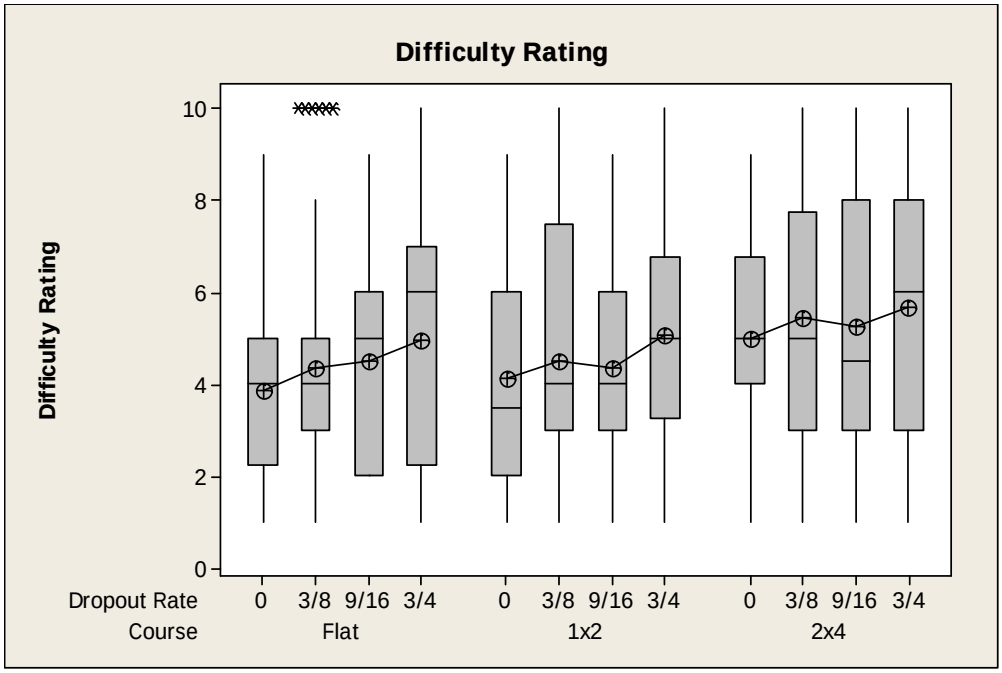

Figure 7: Box plot of difficulty rating as a function of dropout rate and course roughness.

```
                          Individual 95% CIs For Mean Based on Pooled StDev
course_number     Mean     +---------+---------+---------+---------
0              4.41667     (-----*-----)
1              4.51563        (-----*-----)
2              5.33854                                (-----*----)
                           +---------+---------+---------+---------
                          4.20      4.55      4.90      5.25
                                                   I

                         Individual 95% CIs For Mean Based on
                         Pooled StDev
dropout_rate      Mean   ---+---------+---------+---------+------
0              4.33796   (------*------)
1              4.76157              (------*------)
2              4.69907             (-----*------)
3              5.22917                            (-----*------)
                         ---+---------+---------+---------+------
                          4.20      4.55      4.90      5.25
```

Figure 8: The mean difficulty ratings for course roughness and dropout rates.

Table 2 shows the results of a two-way ANOVA of difficulty rating as a function of course roughness and control method ($F_{2,1716}$ = 26.46, p=0.0000). There was significance in the interaction between control method and course roughness. The mean difficulty rating for visual dead reckoning was the lowest of all of the control methods at 4.03. Figure 9 shows the box plot of the two-way ANOVA

results. Figure 10 shows the mean values of the difficulty ratings by course roughness and control method. The mean value of the difficulty rating increased as the course became rougher.

Two-way ANOVA: difficulty_rating_1easy_ versus course_number, control_method

Source	DF	SS	MS	F	P
course_number	2	295.1	147.536	26.46	0.000
control_method	3	664.4	221.472	39.72	0.000
Interaction	6	193.6	32.259	5.78	0.000
Error	1716	9568.9	5.576		
Total	1727	10721.9			

S = 2.361 R-Sq = 10.75% R-Sq(adj) = 10.18%

Table 2: Two-way ANOVA results of difficulty rating as a function of course roughness and control method.

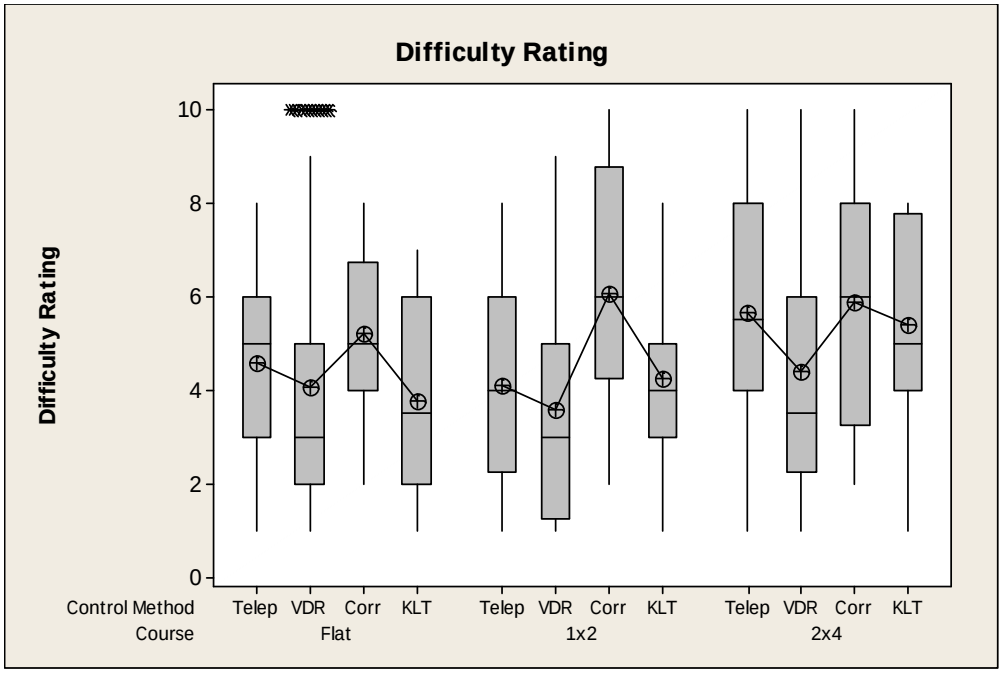

Figure 9: Box plot of difficulty rating as a function of control method and course roughness.

```
                        Individual 95% CIs For Mean Based on
                        Pooled StDev
course_number     Mean  ---------+---------+---------+---------+
0              4.41667  (----*-----)
1              4.51563     (----*-----)
2              5.33854                            (-----*----)
                        ---------+---------+---------+---------+
                               4.55      4.90      5.25      5.60
```

```
                        Individual 95% CIs For Mean Based on
                        Pooled StDev
control_method    Mean  -------+---------+---------+---------+--
0               4.79167              (---*---)
4               4.02778  (---*---)
5               5.72222                              (--*---)
6               4.48611          (---*--)
                        -------+---------+---------+---------+--
                             4.20      4.80      5.40      6.00
```

Figure 10: The mean difficulty ratings for course roughness and control method.

Table 3 shows the results of a two-way ANOVA of difficulty rating as a function of control method and dropout rate ($F_{3,1712}$ = 39.04, p=0.0000). The difficulty rating for visual dead reckoning was the lowest across all courses and control methods. There was significance in the interaction between control method and dropout rate. Figure 11 shows the box plot of the two-way ANOVA results. Figure 12 shows the mean values of the difficulty ratings by control method and dropout rate.

Two-way ANOVA: difficulty_rating_1easy_ versus control_method, dropout_rate

Source	DF	SS	MS	F	P
control_method	3	664.4	221.472	39.04	0.000
dropout_rate	3	173.6	57.875	10.20	0.000
Interaction	9	171.1	19.011	3.35	0.000
Error	1712	9712.8	5.673		
Total	1727	10721.9			

S = 2.382 R-Sq = 9.41% R-Sq(adj) = 8.62%

Table 3: Two-way ANOVA results of difficulty rating as a function of dropout rate and control method.

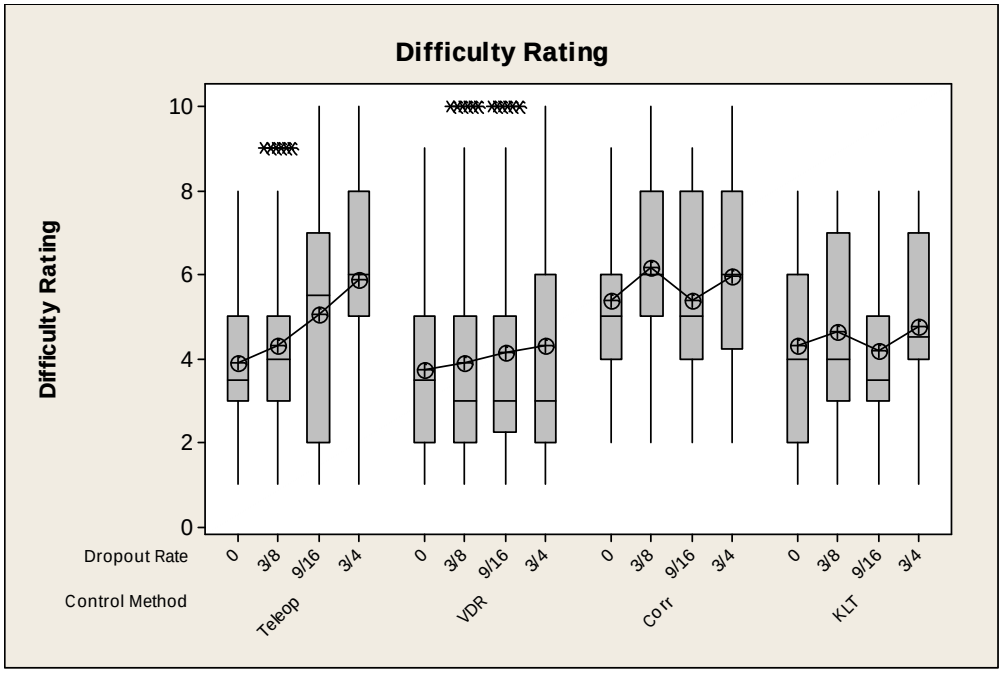

Figure 11: Box plot of difficulty rating as a function of dropout rate and control method.

```
                                Individual 95% CIs For Mean Based on
                                Pooled StDev
control_method      Mean        -------+---------+---------+---------+--
0                4.79167                   (---*---)
4                4.02778        (---*---)
5                5.72222                                   (--*---)
6                4.48611                (---*---)
                                -------+---------+---------+---------+--
                                     4.20      4.80      5.40      6.00

                              Individual 95% CIs For Mean Based on
                              Pooled StDev
dropout_rate      Mean        --+---------+---------+---------+-------
0              4.33796        (-----*-----)
1              4.76157                  (-----*-----)
2              4.69907                (-----*------)
3              5.22917                                (-----*------)
                              --+---------+---------+---------+-------
                              4.20      4.55      4.90      5.25
```

Figure 12: The mean difficulty ratings for control method and dropout rate.

5. Conclusions

The most significant factor found was the difficulty rating. The mean difficulty rating for visual dead reckoning was less than every other control method, which indicates that the subjects found it to be easier than teleoperation. The visual

servoing methods were close and teleoperation was last, especially as the difficulty of terrain and dropout rates increased.

The subject testing took an average of four to five hours to complete. All participants were able to complete the test in one block of time. Generally speaking, the r^2 values were all low. This indicated that there was noise that was not accounted for. This may be due to the fact that participants served as replicates. This could also be due to the operators not feeling comfortable with the control methods. The lighting in the laboratory could not be controlled and it could have caused the visual servoing algorithms to not perform as well as they could.

The sound of the robot when it is operating was very loud in the laboratory setup. Although the subject was positioned in such a way that the robot was not visible at any time during the test, it would be a different experience if the subject was operating the robot in another room where the robot could not be heard as readily.

This did prove that visual dead reckoning was the preferred and most robust of the semi-autonomous algorithms. This also proved that the visual servoing algorithms, as implemented in this research, may not be robust enough for adoption by the Army. The laboratory setting was a benign environment compared to the missions that these robots are required to operate in. If they do not perform well in this setting, it is logical to conclude they won't perform well in Iraq and Afghanistan.

References

1. Chen, J., E. Haas, and M. Barnes, *Human performance issues and user interface design for teleoperated robots*. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 2007. **37**(6): p. 1231-1245.

2. *BP robots still trying to contain oil spill*. 2010-09-11]; Available from: http://www.cbc.ca/world/story/2010/06/02/gulf-of-mexico-oil-spill.html.

3. *BP oil spill containment stalled as robotic saw becomes stuck*. [cited 2010-09-11; Available from: http://www.examiner.com/world-news-in-national/bp-oil-spill-containment-stalled-as-robotic-saw-becomes-stuck.

4. Gunderson, J. and L. Gunderson, *Autonomy (What's it Good for?)*.

5. Sarter, N., D. Woods, and C. Billings, *Automation surprises.* Handbook of human factors and ergonomics, 1997. **2**: p. 1926–1943.

6. Endsley, M., *Toward a theory of situation awareness in dynamic systems.* Human Factors: The Journal of the Human Factors and Ergonomics Society, 1995. **37**(1): p. 32-64.

7. Endsley, M., *Situation awareness in aviation systems.* Handbook of aviation human factors, 1999: p. 257-276.

8. Clapper, J., et al., *Unmanned Systems Roadmap 2007-2032.* 2007.

9. Hutchinson, S., G. Hager, and P. Corke, *A tutorial on visual servo control.* IEEE transactions on robotics and automation, 1996. **12**(5): p. 651-670.

10. Perkins, W. and T. Binford, *A corner finder for visual feedback.* Computer Graphics and Image Processing, 1973. **2**(3-4): p. 355-376.

11. Kragic, D. and H. Christensen, *Survey on visual servoing for manipulation.* Computational Vision and Active Perception Laboratory, Fiskartorpsv. **15**.

12. Wilson, W., C. Hulls, and G. Bell, *Relative end-effector control using Cartesian position based visual servoing: Special section on vision-based control of robot manipulators.* IEEE Transactions on Robotics and Automation, 1996. **12**(5): p. 684-696.

13. Espiau, B., *Effect of camera calibration errors on visual servoing in robotics.* Experimental Robotics III, 1994: p. 182-192.

14. Corke, P., *Visual control of robot manipulators {a review.* Visual Serving: Real Time Control of Robot Manipulators Based on Visual Sensory Feedback, 1993: p. 1.

15. Tsai, R., *A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses.* IEEE Journal of robotics and Automation, 1987. **3**(4): p. 323-344.

16. Espiau, B., F. Chaumette, and P. Rives, *A new approach to visual servoing in robotics.* Geometric Reasoning for Perception and Action, 1993: p. 106-136.

17. Martinet, P. and J. Gallice. *Position based visual servoing using a nonlinear approach*.

18. Malis, E., F. Chaumette, and S. Boudet, *2 1/2 D visual servoing.* IEEE Transactions on Robotics and Automation, 1999. **15**(2): p. 238-250.

19. Bowditch, N., *Dead Reckoning.* The American Practical Navigator, an Epitome of Navigation, 1802.

20. Gallistel, C., *The organization of learning*. 1990: MIT press Cambridge, MA.

21. Pantel, L. and L. Wolf. *On the suitability of dead reckoning schemes for games*. 2002: ACM.